

Appl. No. 10/032,156
 Amdt. dated February 28, 2006
 Response to Notice of Allowance December 30, 2005

PATENT

Amendments to the Specification:

Please replace paragraph [130] with the following amended paragraph:

--[130] Hamming decoder 1210 is also coupled to receive input symbols and redundant symbols from the reconstruction buffer 1215. Additionally, Hamming decoder 1210 receives the number K of input symbols, and the number D, where D+1 is the number of redundant Hamming symbols. Hamming decoder 1210 attempts to recover those input symbols not recovered by the dynamic decoder and the LDPC decoder [2005]1205. While the aim of LDPC decoder [2005]1205 is to recover as many as possible input and redundant symbols, Hamming decoder [2010]1210 only attempts to recover the input symbols IS(0), IS(1), ..., IS(K-1).--

Please replace paragraph [136] with the following amended paragraph:

--[136] Dynamic matrix generator 1305 and static matrix generator 1310 will now be described in further detail with reference to dynamic encoder 500 of Fig. 5 and static encoder [205]210 in Fig. 2. Fig. 18 is a simplified flow diagram illustrating one embodiment of a method employed by dynamic matrix generator 1305. In step 1405, dynamic matrix generator 1205 initializes a matrix C of format (K+A) x (K+R) to all zeros. Next, in step 1410, the keys I_a, I_b, ... are used in conjunction with weight selector 510 and associator 515 to generate the weights W(0), ..., W(K+A-1), and the lists AL(0), ..., AL(K+A-1), respectively. Each of the lists AL(k) comprises W(k) integers in the range 0, ..., K+R-1. In step 1415, these integers are used to compute C(k,l): Where AL(k)=(a(0), ..., a(W(k)-1)), the entries C(k,a(0)), ..., C(k,a(W(k)-1)) are set to 1. As explained above, matrix C gives rise to a system of equations for the unknowns (IS(0), ..., IS(K-1), RE(0), ..., RE(R-1)) in terms of the received symbols (B(0), ..., B(K+A-1)). The reason is the following: once dynamic encoder chooses weight W(k) and associate list AL(k)=(a(0), ..., a(W(k)-1))), the corresponding output symbol B(k) is obtained as

$$B(k) = L(a(0)) \oplus L(a(1)) \oplus \dots \oplus L(a(W(k)-1)),$$

wherein L(j) denotes the unknown value of reconstruction buffer 1925 at position j. These equations, accumulated for all values of k between 0 and K+A-1, give rise to the desired system of equations.--

Appl. No. 10/032,156

PATENT

Amdt. dated February 28, 2006

Response to Notice of Allowance December 30, 2005

Please replace paragraph [142] with the following amended paragraph:

--[142] Another embodiment of a method for implementing an associator 520 for which N need not be a prime number is shown in Fig. 20. First, in a step 1805, a variable k is initialized to zero. Then, in a step 1810, a random integer Y is generated. In one specific embodiment, the key I for the output symbol is used to seed a random number generator. Then, in step 1815, the integer Y is taken modulo the number N to produce a number between 0 and $N-1$. In step 1820, the candidate number Y is tested against other numbers Y previously generated ($X(0)$, $X(1)$, ...). If the number Y had been previously generated, then the flow returns to step 1810. Otherwise, in step 1825, it is included in a list $X(0)$, $X(1)$. Then, in step 1830, it is determined whether $W(I)$ numbers have been generated. If not, then the flow returns to step 1810. The result of the flow illustrated in Fig. [8]20 is a list of $W(I)$ numbers $X(0)$, $X(1)$, ... $X(W(I)-1)$, where each number X in the list is a unique integer between 0 and $N-1$. Then, in a step [835]1835, the list $AL(I)$ is set as the numbers $X(0)$, $X(1)$, ... $X(W(I)-1)$.--